

## **TIMING SYNCHRONIZATION USING DIRTY TEMPLATES IN ULTRA WIDEBAND (UWB) COMMUNICATIONS**

[0001] This application claims priority from U.S. Provisional Application Serial No. 60/453,659, filed March 8, 2003, the entire content of which is incorporated herein by reference.

### **STATEMENT REGARDING FEDERALLY SPONSORED RESEARCH OR DEVELOPMENT**

[0002] This invention was made with Government support under Subcontract #497420 awarded by the University of Delaware (Army Prime #DAAD19-01-2-011). The Government may have certain rights in the invention.

### **TECHNICAL FIELD**

[0003] The invention relates to communication systems and, more particularly, transmitter and receivers used in wireless communication systems.

### **BACKGROUND**

[0004] In general, a receiver in a wireless communication system does not have a priori knowledge of the physical channel over which the transmitted signal propagates or the time at which a transmitter transmits the signal. Timing synchronization, also known as clock recovery, is the process by which the receiver processes a received signal to determine the precise transition points within the received waveform. In other words, the receiver attempts to “synchronize” or align its clock with the clock of the arriving waveform. This process requires the receiver to estimate or otherwise determine the appropriate “timing offset” of the received signal, i.e., the amount of skew between the transmitter’s clock and that of the arriving waveform.

[0005] Incorrect determination of the timing offset can have detrimental effects on other receiver operations, such as channel estimation, symbol detection, and the like. For example, an incorrect timing offset may cause the received waveform to be sampled at times during

which the waveform is in transition between two symbols resulting in an increased number of symbol detection errors.

[0006] In general, timing synchronization in wireless communication systems typically comprises “peak-picking” the output of a sliding correlator between the received signal and a transmit-waveform template. The receiver uses the output of the sliding correlator to estimate the timing offset.

[0007] Ultra-wideband (UWB) systems transmit information via baseband transmissions with high penetration capability and rich multipath diversity that can be collected with low-complexity RAKE reception. However, the information-bearing waveforms are impulse-like and have low power, which increases the difficulty in achieving accurate and efficient timing synchronization. Additionally, peak-picking the output of a sliding correlator between the received waveform and the transmit-waveform template is suboptimum in the presence of dense multipath, and results in high receiver complexity with slow acquisition times.

[0008] Existing sliding correlator timing synchronization techniques operate with assumptions that are unrealistic in UWB systems. More specifically, many techniques require unrealistic assumptions such as an absence of noise or other interference, absence of multipath, and a priori knowledge of the communication channel. For example, one technique, referred to as coarse bin reversal searching, operates under the assumption that there is no noise. Coarse bin reversal searching estimates the arrival time of the received waveform by dividing each symbol duration into thousands of bins and searching for the bin containing the greatest energy with a sliding correlator. Another technique uses a coded beacon sequence designed in conjunction with a bank of correlators to operate in the absence of multipath. The coded beacon sequence is cross-correlated with the received waveform and estimates the location of the beacon sequence within the received waveform via peak-picking. The location of the beacon sequence is used to estimate the timing offset. Non-data-aided, i.e. blind, timing synchronization techniques operate in the presence of dense multipath but require that there is no time-hopping (TH) within each symbol. Such blind timing synchronization techniques use cross-correlation and rely on dense multipath and the cyclostationarity that arises from the time-hopping restriction. So called “ranging systems” do not use a sliding correlator, but rather use a priori knowledge of the strongest path of the communication channel to estimate the distance between transmitter and receiver.

[0009] Consequently, such conventional timing synchronization techniques do not operate functionally in realistic UWB environments. Moreover, many of these conventional timing synchronization techniques for UWB systems, especially those that utilize a sliding correlator with a transmit-waveform template, result in slow acquisition and high receiver complexity.

## SUMMARY

[0010] In general, techniques are described that provide timing synchronization via correlating received waveforms with “dirty” templates, in wireless communication systems, such as ultra-wideband systems (UWB). Unlike exiting UWB timing synchronization techniques and early-late gate narrowband timing synchronization techniques, the techniques described herein peak-pick the output of a sliding correlator with “dirty templates,” i.e. templates that propagate through the UWB communication channel. These dirty templates are segments of the received waveform and are not only distorted by the unknown channel but also noisy and subject to the unknown timing offset. As shown herein, the use of the dirty templates in timing synchronization may achieve increased energy capture, resulting in improved performance and enabling reduced receiver complexity. Further, the described techniques are able to achieve timing synchronization at any desirable resolution, and are applicable to non-UWB systems provided inter-symbol interference is absent or minimal.

[0011] In one embodiment, the invention is directed to a method which processes a data stream of information-bearing symbols to form bursts of information symbols and arranges training symbols within the stream of information-bearing symbols to ensure one polarity transition within three training symbols at a receiver to synchronize timing of a transmitter and the receiver. The method outputs the bursts of information-bearing symbols and training symbols as an ultra wideband (UWB) waveform through a communication channel to the receiver.

[0012] In another embodiment, the invention is directed to a method which receives an ultra wideband (UWB) waveform through a wireless communication channel, wherein the received UWB waveform comprises bursts of information-bearing symbols. A template is selected to be used for estimating the timing offset of a burst of the received UWB waveform, wherein the template comprises a segment of a burst of the received UWB waveform and the template is correlated with a segment of a burst of the received waveform

so as to form an estimate of the timing offset of the received UWB waveform. A stream of symbol estimates is output in accordance with the estimated timing offset.

**[0013]** In yet another embodiment, the invention is directed to a wireless communication device comprising a pulse generator and a pulse shaping unit. The pulse generator processes a data stream of information bearing symbols to form bursts of information bearing symbols and arranges training symbols within the stream of information-bearing symbols to ensure one polarity transition within three training symbols at a receiver to synchronize timing of a transmitter and the receiver. The pulse shaping unit outputs an ultra wideband (UWB) transmission waveform from the bursts of information-bearing symbols and training symbols.

**[0014]** In an additional embodiment, the invention is directed to a wireless communication device comprising an antenna to receive an ultra wideband (UWB) waveform through a wireless communication channel, wherein the received UWB waveform includes bursts of information-bearing symbols, a timing synchronization unit to form an estimation of a timing offset based on the received UWB waveform, and a symbol detector to output a stream of estimate symbols based on the estimate of the timing offset.

**[0015]** In another embodiment, the invention is directed to a system comprising a transmitter and a receiver. The transmitter processes a data stream of symbols to form bursts of information-bearing symbols and generates an ultra wideband (UWB) waveform through a wireless communication channel. The receiver receives the transmitted signal through a wireless communication channel, selects a segment of the received UWB waveform to use as a template, forms an estimate of the timing offset based on the correlation of the template with the received UWB waveform, and outputs a stream of estimate symbols based on the estimate of the timing.

**[0016]** The timing synchronization techniques described herein may offer one or more advantages. For example, peak-picking the output of a sliding correlator with dirty templates allows timing synchronization in realistic UWB environments, and some non-UWB systems as well, provided intersymbol interference is avoided. In particular, the described techniques operate in the presence of noise, multipath, time-hopping, and multiple users. Additionally, timing synchronization can be achieved at any desirable resolution by selecting the time increment at which each correlation is performed within a symbol duration. Moreover,

voltage controlled clock (VCC) circuits can be used to implement the receiver, which may yield lower-complexity UWB receivers than with conventional techniques.

[0017] Other advantages that may be provided by the described techniques include a selectable trade-off between performance and bandwidth efficiency. Selecting data-aided dirty templates may increase the performance of the described techniques. For example, a dirty template with four training symbols increases the performance by reducing the number of operations required at the receiver to synchronize timing with the transmitter. However, selecting a non-data-aided, template provides a more efficient use of bandwidth.

[0018] The details of one or more embodiments of the invention are set forth in the accompanying drawings and the description below. Other features, objects, and advantages of the invention will be apparent from the description and drawings, and from the claims.

### **BRIEF DESCRIPTION OF DRAWINGS**

[0019] FIG. 1 is a block diagram illustrating a wireless communication system using an ultra-wideband (UWB) format performing timing synchronization with dirty templates.

[0020] FIG. 2 is a block diagram illustrating in further detail the wireless communication system of FIG. 1.

[0021] FIG. 3 is a block diagram illustrating an example embodiment of a timing synchronization unit within a receiver.

[0022] FIG. 4 is a timing diagram illustrating exemplary symbol-long waveform segments observed by a receiver for two separate received waveforms.

[0023] FIG. 5 illustrates an example waveform with pairs of symbols alternating between positive polarity and negative polarity for data-aided timing synchronization with dirty templates.

[0024] FIG. 6 illustrates an exemplary mode of operation of the timing synchronization unit of FIG. 3 using the data-aided timing synchronization with the dirty template waveform of FIG. 5.

[0025] FIG. 7 illustrates the second step of the data-aided timing synchronization of FIG. 7.

[0026] FIG. 8 illustrates the third step of the data-aided timing synchronization of FIG. 7.

[0027] FIG. 9 is a flow chart illustrating an exemplary mode of operation of a communication system in which data-aided timing synchronization is performed with dirty templates.

[0028] FIG. 10 illustrates successive symbol long waveform segments of a communication system employing non-data-aided timing synchronization with dirty templates.

[0029] FIG. 11 illustrates an example mode of operation of a communication system employing data-aided timing synchronization with dirty templates in the presence of multiple users.

[0030] FIGS 12-17 are graphs illustrating results of simulations and comparisons that validate analyses and designs of data-aided and non-data-aided timing synchronization with dirty templates.

## DETAILED DESCRIPTION

[0031] FIG. 1 is a block diagram illustrating ultra wideband (UWB) communication system 2 in which transmitter 4 communicates with receiver 8 through wireless channel 6. The techniques are described in reference to UWB, but are also applicable to any non-UWB system provided inter-symbol interference is avoided or minimized. In general, the invention provides techniques for performing timing synchronization with “dirty templates”, i.e., segments of the received waveform which propagates through channel 6. In the described embodiments, communication system 2 may be subject to noise, multipath, multiple-users, and time-hopping (TH). The dirty template can be selected to be “data-aided,” i.e., selected to include one or more training symbols, or “non-data-aided,” also referred to as “blind.” As shown herein, data-aided dirty templates generally result in higher performance, while blind dirty templates result in a more efficient use of bandwidth.

[0032] Transmitter 4 transmits data as a UWB waveform using a modulation scheme, such as pulse position modulation (PPM) or pulse amplitude modulation (PAM). For simplicity, the examples that follow will be described using antipodal PAM. User separation is achieved via TH codes which time-shift the pulse positions at multiples of the chip duration  $T_c$ . Each information-bearing symbol is transmitted over  $N_f$  frames each of duration  $T_f$ . During each frame, a data modulated ultra short pulse  $p(t)$  that has duration  $T_p \ll T_f$  is transmitted. Each pulse, also referred to as a monocycle, can have a Gaussian, Rayleigh, or other shape used in

wireless communications. At time  $t=t_1$  transmitter 4 transmits the UWB waveform according to equation (1). The energy per monocycle is denoted  $\epsilon$ ,  $s(k)$  is the  $k^{\text{th}}$  symbol, and  $p_T(t)$  denotes the transmitted symbol waveform according to equation (2) with  $N_f$  monocycles shifted by the TH code  $c_{nf}$  during the  $n_f^{\text{th}}$  frame.

$$u(t) = \sqrt{E} \sum_{k=0}^{+\infty} s(k) p_T(t - kT_s - t_1) \quad (1)$$

$$p_T(t) := \sum_{n_f=0}^{N_f-1} p(t - n_f T_f - c_{n_f} T_c) \quad (2)$$

[0033] Receiver 8 receives the transmitted UWB waveform through communication channel 6 which is modeled as a tapped-delay line with  $L+1$  taps with coefficients  $\{\alpha_\ell\}_{\ell=0}^L$  and delays  $\{\tau_\ell\}_{\ell=0}^L$  satisfying  $\tau_\ell < \tau_{\ell+1}$ ,  $\forall \ell$ . As a result, the waveform arriving at the receiver is given by equation (3). Additionally, channel 6 is modeled as quasi-static, i.e., the channel coefficients and delays remain invariant over one transmission burst, but can change independently from burst to burst.

$$r(t) = \sum_{\ell=0}^L \alpha_\ell u(t - \tau_\ell) \quad (3)$$

[0034] Denoting the first arrival time at receiver 8 with  $t_2$ , and the propagation delay through channel 6 as  $\tau_0$ , the first arrival time can be written  $t_2 = t_1 + \tau_0$ . Relative to  $\tau_0$ , other path delays can be uniquely described by  $\tau_{\ell,0} = \tau_\ell - \tau_0$ . Consequently, the aggregate channel, i.e. channel 6 in combination with the pulse shaper of transmitter 4, is given by equation (4). Equation (5) expresses the received waveform where the received symbol waveform is defined according to equation (6). The noise term  $w(t)$  includes additive white Gaussian noise (AWGN) and multiple user interference (MUI). Inter-symbol interference is avoided by selecting  $T \geq \tau_{L,0} + T_p$  and  $c_0 = c_{N_f-1} = 0$  which confines  $p_T(t)$  over an interval of duration  $[0, T_s)$ .

$$h(t) := \sum_{l=0}^L \alpha_l p(t - \tau_{l,0}) \quad (4)$$

$$r(t) = \sqrt{E} \sum_{k=0}^{+\infty} s(k) p_R(t - kT_s - t_2) + w(t) \quad (5)$$

$$p_R(t) = \sum_{l=0}^L \alpha_l p_T(t - \tau_{l,0}) \quad (6)$$

[0035] The received waveform  $r(t)$  is received by receiver 8 at time  $t_2$ , but receiver 8 does not know transmission time  $t_1$  or propagation delay  $\tau_0$ . Therefore, receiver 8 initiates recovery techniques, such as timing synchronization, channel estimation and symbol detection, at time  $t_3$  when a change in the energy of received waveforms is detected. Receiver 8 initiates recovery techniques at time  $t_3$  after the arrival of the first received waveform at time  $t_2$  where  $t_3 > t_2$ . The timing offset may be viewed as the difference in time between  $t_3$  and the starting time of a symbol at or after  $t_3$ . The first symbol starting after  $t_3$  is received at time  $t$  defined in equation (7). The arrival time  $t_2$  serves as reference while time  $t_1$  plays no role. As a result,  $t_2$  is set to zero without loss of generality.

$$t = t_2 + \lceil (t_3 - t_2) / T_s \rceil T_s \quad (7)$$

[0036] Equations (8, 9) allow timing synchronization to be corrected by an integer number  $n_\varepsilon \in [0, N_f - 1]$  of frames such that  $t_3 + n_\varepsilon T_f$  is close to either  $NT_s$  or  $(N+1)T_s$  within one symbol duration.

$$N := t_3 / T_s \quad (8)$$

$$\varepsilon := t_3 - NT_s \quad (9)$$



[0037] Data-aided dirty templates based on a finite number of training symbols must satisfy equation (10) where  $K$  is the total number of training symbols, i.e. preamble, and  $K_0$  is the minimum number of training symbols required to implement the described data-aided timing synchronization with dirty templates.

$$N \leq K - K_0 \quad (10)$$

[0038] The techniques described herein may be applied to uplink and/or downlink UWB transmissions, i.e., transmissions from a base station to a mobile device and vice versa. Consequently, transmitter 4 and receiver 8 may be any device configured to communicate using a wireless transmission including a distribution station, a hub for a wireless local area network, a mobile phone, a laptop or handheld computing device, a personal digital assistant (PDA), a device within a wireless personal area network, a device within a sensor network, or other device.

[0039] FIG. 2 is a block diagram illustrating in further detail transmitter 4 and receiver 8 of the exemplary wireless communication system 2 of FIG. 1. In general, pulse generator 10 processes data stream 9 of information-bearing symbols to generate a burst of symbols. In some embodiments, pulse generator 10 inserts training symbols into each burst of symbols. For example,  $K$  training symbols may be inserted as a preamble to each burst. Pulse shaping unit 12 outputs a stream of short pulses to form UWB transmission waveform 11 as given in equation (1) from the burst of symbols. Transmission waveform 11 is a continuous-time waveform carrying the pulse-shaped pulses through channel 6. Channel 6 provides multiple paths for transmission waveform 11, and noise  $\eta(t)$  14 is modeled as AWGN and includes MUI.

[0040] The received UWB waveform 15 is given according to equation (5). Receiver 8 does not know when transmitter 4 began transmitting or the propagation delay of channel 6. Consequently, receiver 8 timing synchronization unit 14 turns on when a change in energy is detected in received waveforms 15. Timing synchronization unit 14 operates on segments of received waveform 15 of duration  $T_s$  starting at time  $(t_3 + nT_f + mT_s)$  for integers  $n \in [0, N_f]$  and  $m \in [0, M-1]$  with  $MT_s$  being the observation interval. Equation (11) expresses this waveform with noise ignored for simplicity. The noise free received wave form can then be

expressed according to equation (12) using the definition of  $t_3$  in equation (13) and defining the aggregate delay, i.e. delay through the aggregate channel, in equation (14).

$$\chi_{n,m}(t) = x(t + mT_s + nT_f + t_3), \quad t \in [0, T_s) \quad (11)$$

$$\chi_{n,m}(t) = \sqrt{E} \sum_{k=0}^{+\infty} s(k) \sum_{n_f=0}^{n_f-1} \sum_{l=0}^L \alpha_l p(t - D_{k,n,m}) \quad (12)$$

$$t_3 = NT_s + \varepsilon \quad (13)$$

$$D_{k,n,m} = (k - N - m)T_s - (nT_f + \varepsilon) \quad (14)$$

[0041] Because  $p(t)$  has finite support over the interval  $(0, T_p)$  only a finite number of  $k$  values contribute non zero summands in equation (10). The  $k$  values are determined by equations (15, 16)

$$\varepsilon + nT_f \in [0, T_s) \Rightarrow k = N + m + [0,1] \quad (15)$$

$$\varepsilon + nT_f \in [T_s, 2T_s) \Rightarrow k = N + m + [1,2] \quad (16)$$

[0042] Expressing  $(nT_f + \varepsilon)$  as an integer multiple of  $T_s$  plus a residue gives equations (17, 18) respectively where  $n_0$  has possible values of 0 and 1 and  $\varepsilon_0$  has possible values of  $[0, T_s)$ .

$$n_0 := (\varepsilon + nT_f) / T_s \quad (17)$$

$$\varepsilon_0 := \varepsilon + nT_f - n_0 T_s \quad (18)$$

[0043] Using equations (17, 18) equations (15, 16) give  $k = N + m + n_0 + [0,1], \forall \varepsilon_0 \in [0, T_s)$ . Consequently equation (12) can be rewritten according to equation (19).

$$\chi_{n,m}(t) = \sum_{k=0}^1 s(k + N + m + n_0) p_R(t - kT_s + \varepsilon_0) \quad (19)$$

[0044] The  $p_R(t - kT_s + \varepsilon_0)$  depends only on  $\varepsilon$  and  $n$ , but is independent of  $N$ ,  $m$ , and  $n_0$ . In other words, as  $m$  changes  $p_R(t - kT_s + \varepsilon_0)$  remains invariant for any given  $n$  and  $\varepsilon$ .

[0045] Timing synchronization unit 14 performs a cross correlation between successively observed waveforms  $x_{n,m}(t)$ , each of duration  $T_s$ , to acquire  $n_\epsilon$ . Integrating over one symbol duration  $T_s$ , the correlation between successive waveforms  $x_{n,m}(t)$  and  $x_{n,m+1}(t)$  is given by equation (20). From equation (20) the timing synchronization with data-aided dirty templates is designed below with the description of blind time synchronization using dirty templates following.

$$\begin{aligned}
 R_{xx}(n;m) &:= \int_0^{T_s} \chi_{n,m}(t) \chi_{n,m+1}(t) dt \\
 &= E \int_0^{T_s} \sum_{k=0}^1 s(k+N+m+n_0) p_R(t-kT_s+\epsilon_0) \\
 &\quad \times \sum_{k=0}^1 s(k+N+m+n_0+1) p_R(t-kT_s+\epsilon_0) dt
 \end{aligned} \quad (20)$$

[0046] Timing synchronization unit 14 uses data-aided dirty templates to increase performance of estimating the timing offset. In some embodiments, training symbols are transmitted at the beginning of each burst to enable timing synchronization unit 14 to estimate the timing offset. As a result, the recovery techniques performed after such timing synchronization will not be substantially affected by the timing offset. Transmitting training symbols reduces the bandwidth of the UWB system, but increases the performance. Therefore, minimizing the number of training symbols results in an optimal trade-off between performance and bandwidth efficiency in data-aided timing synchronization. The techniques described here allow a minimum number of four symbols to be used. In some embodiments, the minimum number of symbols used is four.

[0047] In order to find the minimum number of training symbols, equation (20) is rewritten as equation (21) with coefficients A, B, and C defined in equations (22, 23) respectively. In order to minimize the number of training symbols, which results in a low complexity implementation of receiver 8, the dependence of  $R_{xx}(n; m)$  on N, m, and  $n_0$  is removed.

$$R_{xx}(n; m) = A \int_0^{T_s} p_R^2(t + \varepsilon_0) dt + B \int_0^{T_s} p_R^2(t - T_s + \varepsilon_0) dt \quad (21)$$

$$A := E \bullet s(N + m + n_0) s(N + m + n_0 + 1) \quad (22)$$

$$B := E \bullet s(N + m + n_0 + 1) s(N + m + n_0 + 2) \quad (23)$$

[0048] Using the training sequence given by equation (24) and using equations (22,23) the possible values of A, B, and C are given in Table (1). Therefore, the data-aided dirty template is a preamble training sequence of two positive pulses followed by two negative pulses of unit amplitude that is received by receiver 8. Receiver 8 observes successive waveforms over a symbol duration as previously described. Due to the timing offset, the symbol duration long waveforms will include a fraction of two training symbols. However, receiver 8 incrementally correlates successive waveforms over a symbol duration and peak-picks the increment with the greatest result.

$$s(k) = \begin{cases} +1, & \text{if } (k \bmod 4) = 0, \text{ or } 1 \\ -1, & \text{if } (k \bmod 4) = 2, \text{ or } 3 \end{cases} \quad (24)$$

[0049] In some embodiments, the size of the increment is adjusted to achieve coarse timing synchronization, also referred to as acquisition, by incrementing at the frame level to achieve an integer multiple of the symbol duration, i.e.  $n_s$ , with ambiguity less than one frame duration. In other embodiments, the size of the increment is adjusted to achieve fine timing synchronization, also referred to as tracking, by incrementing at the chip level to achieve an integer multiple of the frame duration. In further embodiments, the size of the increment is a non-integer value and timing synchronization unit 16 is implemented with voltage controlled clock (VCC) circuitry.

$s(N+m+n_0)$	+1	+1	-1	-1
$s(N+m+n_0+1)$	+1	-1	-1	+1
$s(N+m+n_0+2)$	-1	-1	+1	+1
A	+E	- E	+ E	- E
B	- E	+ E	- E	+ E

Table 1

[0050] Table 1 explicitly shows all possible values of A, B, and C. The absolute value of  $R_{xx}(n; m)$  remains invariant for all  $m$ ,  $N$ , and  $n_0$  and is given by equation (25) where the last equality is obtained through a change of variables. Because the symbol waveform  $p_R(t)$  contains  $N_f$  monocycles, the first integral corresponds to monocycles with indices from  $\lfloor \varepsilon_0/T_f \rfloor - 1$  to  $N_f - 1$  while the second integral corresponds to monocycles with indices from 0 to  $\lfloor \varepsilon_0/T_f \rfloor$ . In the absence of TH codes, i.e.  $c_{nf}=0$ ,  $\forall n_f \in [0, N_f - 1]$ , the two integrals in equation (25) can be expressed in closed form respectively as given in equations (26, 27) where  $E_h(\tau)$  is given in equation (28) and  $\varepsilon_1$  is given in equation (29).

$$R_{xx}(n) := |R_{xx}(n; m)| = E \bullet \left| \int_0^{T_s} p_R^2(t + \varepsilon_0) dt - \int_0^{T_s} p_R^2(t - T_s + \varepsilon_0) dt \right|$$

$$= E \bullet \left| \int_{\varepsilon_0}^{T_s} p_R^2(t) dt - \int_0^{\varepsilon_0} p_R^2(t) dt \right| \quad (25)$$

$$\int_{\varepsilon_0}^{T_s} p_R^2(t) dt = \left( N_f - \left\lfloor \frac{\varepsilon_0}{T_f} \right\rfloor \right) E_h(T_f) - E_h(\varepsilon_1) \quad (26)$$

$$\int_0^{\varepsilon_0} p_R^2(t) dt = \left\lfloor \frac{\varepsilon_0}{T_f} \right\rfloor E_h(T_f) + E_h(\varepsilon_1) \quad (27)$$

$$E_h(\tau) := \int_0^\tau h^2(t) dt \quad (28)$$

$$\varepsilon_1 := (\varepsilon_0 \bmod T_f) \quad (29)$$

[0051] However, in the presence of TH codes, the two integrals in equations (25) cannot be expressed in closed form but the two integrals do not vary much with and without TH codes

for various  $\varepsilon_0$  values and channel realizations as will be shown in greater detail below. Substituting equations (26, 27) into equation (25) gives equation (30).

$$R_{xx}(n) = \varepsilon \bullet \left( N_f - 2 \left\lfloor \frac{\varepsilon_0}{T_f} \right\rfloor \right) \varepsilon_h(T_f) - 2\varepsilon_h(\varepsilon_1) \quad (30)$$

**[0052]** The maximum of  $R_{xx}(n)$  of equation (30) occurs at  $\lfloor \varepsilon_0/T_f \rfloor = 0$ . Consequently, for any  $\varepsilon$ , a unique  $n_\varepsilon$  can be found by peak-picking  $R_{xx}(n)$  across  $n$  as shown in equation (31) by using equation (18) and noticing that  $(\varepsilon + nT_f) \in [0, 2T_s)$ .

$$n_\varepsilon := \arg \max_n \{R_{xx}(n)\} = N_f - \left\lfloor \frac{\varepsilon}{T_f} \right\rfloor \quad (31)$$

$$t_3 + n_\varepsilon T_f = NT_s + \varepsilon + (N_f - \lfloor \varepsilon/T_f \rfloor)T_f = (N+1)T_s + (\varepsilon \bmod T_f) = (N+1)T_s + \varepsilon_1 \quad (32)$$

**[0053]** Equation (32) shows that equation (31) leads to estimating the timing offset by correcting  $t_3$  with  $n_\varepsilon T_f$ , which results in acquiring  $(N+1)T_s$ . Further, because  $\varepsilon_1 \in [0, T_f)$ , the acquisition error is bounded by one frame duration, i.e. frame-level resolution. In other words, timing synchronization using dirty template with training symbols is achieved by peak-picking the absolute value of the cross correlation in equation (30) between successive waveforms  $x_{n,m}(t)$  each of duration  $T_s$ . Correcting the starting time  $t_3$  of timing synchronization unit 14 with an estimate obtained from equation (31) yields starting time  $(N+1)T_s$  with error less than  $T_f$ .

**[0054]** Additionally, if  $\varepsilon_1$  is uniformly distributed over the interval  $[0, T_s)$  then  $\varepsilon_1$  is uniformly distributed over the interval  $[0, T_f)$ . As a result,  $E\{t_3 + n_\varepsilon T_f | t_3\} = (N+1)T_s + T_f/2$ . An estimate is obtained by computing  $R_{x,x}(n;m)$  over pairs of  $x_{n,m}(t)$  and averaging their absolute values across the  $M/2$  pairs as given in equation (33).

$$\hat{R}_{xx}(n) = \frac{2}{M} \sum_{m=0}^{M/2-1} \left| \int_0^{T_s} x_{n,2m}(t) x_{n,2m+1}(t) dt \right| \quad (33)$$

[0055] In order to carry out timing synchronization with a data-aided dirty template,  $M$  must at least be 2, i.e.  $m=[0, 1]$ . From equations (15, 16)  $m=1$  implies  $k \geq N+3$ . Therefore, the minimum number of training symbols  $K$  must satisfy  $K-1 \geq N+3$ . In other words,  $N$  has to be bounded by  $K-4$  for any given number of training symbols  $K$ .

[0056] Timing synchronization using a non-data-aided template trades-off performance for efficient bandwidth. All pulses in non-data-aided timing synchronization are information-bearing, resulting in efficient use of bandwidth. However, longer observation intervals are required to obtain the same performance as data-aided timing synchronization because the information symbols are not guaranteed to switch from positive to negative within two symbols as was designed in the data-aided technique above.

[0057] Blind timing synchronization using dirty templates follows from equation (20).

Equation (20) can be rewritten as equation (34) because  $\int_0^{T_s} p_R(t + \varepsilon_s) p_R(t + T_s + \varepsilon_s) dt$  vanishes due to the finite support of  $p_R(t)$  and substitutions  $\int_{\varepsilon_s}^{T_s} p_R^2(t) dt = \int_0^{T_s} p_R^2(t + \varepsilon_s) dt$  and  $\int_0^{\varepsilon_s} p_R^2(t) dt = \int_0^{T_s} p_R^2(t - T_s + \varepsilon_s) dt$ . Taking the absolute value of  $R_{xx}(n; m)$  removes the dependence on the unknown symbol  $s(N + m + n_s + 1)$  and gives equation (35).

$$R_{xx}(n; m) = E \bullet s(N + m + n_s + 1) \left[ s(N + m + n_s) \int_{\varepsilon_s}^{T_s} p_R^2(t) dt + s(N + m + n_s + 2) \int_0^{\varepsilon_s} p_R^2(t) dt \right] \quad (34)$$

$$|R_{xx}(n; m)| = E \bullet \left| s(N + m + n_s) \int_{\varepsilon_s}^{T_s} p_R^2(t) dt + s(N + m + n_s + 2) \int_0^{\varepsilon_s} p_R^2(t) dt \right| \quad (35)$$

[0058] Moreover, with information conveying symbols  $s(k) \in \{\pm 1\}$  being independent and taking the conditional expectation of the absolute value  $|R_{xx}(n; m)|$  for any  $n \in [0, N-1]$  gives equation (36).

$$R_{xx}(n) := E \left\{ |R_{xx}(n; m)| \middle| h(t), \varepsilon \right\} = \frac{E}{2} \left| \int_{\varepsilon_s}^{T_s} p_R^2(t) dt - \int_0^{\varepsilon_s} p_R^2(t) dt \right| + \frac{E}{2} \int_0^{T_s} p_R^2(t) dt \quad (36)$$

[0059] Equation (36) removes the dependence of the absolute value  $|R_{xx}(n;m)|$  on  $N$ ,  $m$ , and  $n_s$ . However,  $R_{xx}(n)$  is still dependent on  $n$  for any timing offset  $\varepsilon$ .

[0060] As described previously with data-aided timing synchronization, the symbol waveform  $p_R(t)$  contains  $N_f$  monocycles. As a result, the first integral corresponds to monocycles with indices from  $\lfloor \varepsilon_0/T_f \rfloor - 1$  to  $N_f - 1$  while the second integral corresponds to monocycles with indices from 0 to  $\lfloor \varepsilon_0/T_f \rfloor$ . Similarly, in the absence of TH codes, i.e.  $c_{mf}=0$ ,  $\forall n_f \in [0, N_f - 1]$ , the two integrals in equation (26) can be expressed in closed form respectively as given in equations (27, 28) where  $E_h(\tau)$  is given in equation (29) and  $\varepsilon_1$  is given in equation (30).

[0061] Substituting equations (27, 28) into equation (36) gives equation (37) because  $\varepsilon_s \in [0, T_s]$ ,  $\lfloor \varepsilon_s/T_f \rfloor$  is in the interval  $[0, N_f - 1]$ . Consequently, the maximum of  $R_{xx}(n)$  in equation (37) is given by equation (38).

$$R_{xx}(n) = \frac{E}{2} \left( N_f - 2 \left\lfloor \frac{\varepsilon_0}{T_f} \right\rfloor \right) E_h(T_f) - 2E_h(\varepsilon_1) + \frac{E}{2} N_f E_h(T_f) \quad (37)$$

$$\max_n \{R_{xx}(n)\} = EN_f E_h(T_f) - E \min \{E_h(\varepsilon_1), E_h(T_f) - E_h(\varepsilon_1)\} \quad (38)$$

[0062] The maximum occurs at  $\lfloor \varepsilon_s/T_f \rfloor = 0$  if  $E_h(\varepsilon_1) < E_h(T_f)/2$  and at  $\lfloor \varepsilon_0/T_f \rfloor = N_f - 1$  otherwise. Using the definition of  $\varepsilon_0$  and noticing that  $n \in [0, N_f - 1]$  gives piece-wise function (39).

$$\arg \max_n \{R_{xx}(n)\} = -\lfloor \varepsilon/T_f \rfloor = 0 \text{ when } \varepsilon = 0$$

and

$$\arg \max_n \{R_{xx}(n)\} = \begin{cases} ((N_f - \lfloor \varepsilon/T_f \rfloor) \bmod N_f) \\ N_f - \lfloor \varepsilon/T_f \rfloor - 1 \end{cases} \text{ if } \begin{cases} E_h(\varepsilon_1) < E_h(T_f)/2 \\ \text{otherwise} \end{cases} \text{ when } \varepsilon > 0 \quad (39)$$

[0063] Equation (39) peak picks  $R_{xx}(n)$  across  $n$  for any  $\varepsilon$ . In other words,  $\arg \max_n \{R_{xx}(n)\}$  gives  $n_\varepsilon$  because by correcting  $t_3$  with the number of frames given by equation (39),



either  $NT_s$  or  $(N + 1)T_s$  is acquired with ambiguity  $< T_f$ . Table 2 explicitly lists the possible values of  $t_3 + \arg \max_n \{R_{xx}(N)\}T_f$ .

	$E_h(\epsilon_1) < \epsilon_h(T_f)/2$	$E_h(\epsilon_1) > E_h(T_f)/2$
$\epsilon=0$	$NT_s$	Impossible
$\epsilon \in (0, T_f)$	$NT_s + \epsilon_1$	$(N + 1)T_s + (\epsilon_1 - T_f)$
$\epsilon \in [T_f, T_s)$	$(N + 1)T_s + \epsilon_1$	$(N + 1)T_s + (\epsilon_1 - T_f)$

Table 2

[0064] Similar to data-aided timing synchronization, equation (33) estimates  $R_{xx}(n;m)$  by computing  $R_{xx}(n;m)$  over pairs of  $x_{n,m}(t)$  each of duration  $T_s$ , and averaging the resulting values across the  $M/2$  pairs. Peak-picking equation (33), i.e.  $n'_\epsilon = \arg \max_n \{\hat{R}_{xx}(n)\}$  allows  $t_3$  to be corrected with  $\hat{n}_\epsilon T_f$  results and results in an estimate of a symbol starting time. Equation (33) is not limited to an integer output and synchronization unit 14 can be implemented with VCC circuits.

[0065] Timing synchronization unit 14 estimates the timing offset and initiates channel estimator 16 at the start of the first symbol of received waveform 15 after  $t_3$  with resolution determined by the number of increments. Channel estimator 16 reduces the effects of channel 6, including attenuation and multipath effects. Matched filter 17 receives the output of channel estimator 16 and separates multiple users of communication system 2. Sampling unit 18 samples the output of matched filter 17 producing an output of symbols. Symbol detector 19 performs symbol detection on the output of matched filter 17 producing a stream of symbol estimates 20.

[0066] FIG. 3 is a block diagram illustrating an example embodiment of timing synchronization unit 14 of receiver 8 (FIG. 2). For purposes of illustration, timing synchronization unit 14 employs data-aided timing synchronization. Clock 22 receives waveform 15 and initiates data-aided timing synchronization when a change in amplitude of received waveforms 15 is detected. Clock 22 multiplies the received waveform by delay  $T_s$  and correlator 24 correlates  $M/2$  successive symbol duration waveform pairs. via equations (30) starting at time  $t_3$ . Clock 22 multiplies  $t_3$  by a selectable value, such as a frame duration,

i.e.,  $t_3' = t_3 + T_f$ , until  $t_3$  is incremented by a symbol duration  $t_3' = t_3 + T_s$ . Correlator 24 correlates  $M/2$  successive symbol duration waveform pairs for each incremented time value  $t_3'$ . Decision unit 26 receives the output of correlator 24 for each time value and selects the increment value that resulted in the greatest correlation according to equation (30). Equation (24) employs a training sequence such that the greatest correlation occurs when the  $M/2$  pairs of waveforms are each of opposite polarity. Equation (18) implies that by correcting  $t_3$  with  $n_e T_f$ ,  $(N+1)T_s$  is obtained. Additionally, because  $\varepsilon_1 \in [0, T_f]$ , the acquisition error is bounded by one frame duration. Using similar techniques, fine timing synchronization can be achieved with variable non-integer increments.

[0067] The embodiment of timing synchronization unit 14 is described for exemplary purposes only. A substantially similar timing synchronization unit may be employed for non-data-aided timing synchronization. Such a timing synchronization unit would correlate  $M/2$  successive symbol duration waveforms pairs. However, instead of correlating dirty templates of training symbols, the non-data-aided timing synchronization unit would correlate dirty templates of information symbols. Because training symbols are not being transmitted, non-data-aided timing synchronization results in increased bandwidth efficiency. However, to achieve similar performance as the data-aided timing synchronization technique  $M$  is generally a larger value, resulting in a slower acquisition time and increased complexity.

[0068] FIG. 4 is a timing diagram illustrating exemplary waveform segments of duration  $T_s$  for two separate received waveforms 34 and 36 which are observed by receiver 8 of communication system 2. For illustrative purposes, supplemental notation for describing timing synchronization with dirty templates is introduced in the following sections. The notation introduced in relation to the following figures should not be taken as contradictory to the previous notation or as limiting the previous notation. Rather, the supplemental notation should be viewed as supporting the following figures and substantially strengthening a conceptual illustration of timing synchronization with dirty templates.

[0069] For reference, exemplary received symbol 30 with symbol duration  $T_s = N_f T_f$  includes  $N_f = 6$  pulses, each with duration  $T_f$ , transmitted by transmitter 4. Equation (40) expresses exemplary symbol 30 in terms of the aggregate channel given in equation (4). Waveforms 32 and 34 each include two UWB symbols with received pulses represented as triangles to indicate the time spreading multipath effects of channel 6. Positive pulses are represented by

pulses with a solid fill, while negative pulses are represented by pulses with a cross-hatch fill. Waveform 32, denoted  $x_{\tau_0}(t)$  as given in equation (41), includes the pulses, which are all positive, within window 33. Window 33 is of one symbol duration  $T_s$ . Waveform 34, denoted  $y_{\tau_0}(t)$  as given in equation (42), includes the received pulses, positive pulses followed by negative pulses, within window 33. Interval 36 has duration  $T_s$  and begins when receiver 8 detects an increase in the amplitude of received waveforms. The timing offset created by the difference in time between the time at which interval 36 begins and when the next symbol starts is given by  $\tau_0 = n_s T_f + \varepsilon$ .

[0070] Even in the presence TH and unknown multipath, least-squares (LS) optimal timing estimation is given by equation (43). Peak-picking the cross correlation of  $T_s$  long received segments of waveforms 32 and 34 allows timing acquisition and tracking in one step or in separate steps at any desirable resolution. Waveforms 32 and 34 guarantee LS estimation of timing off set  $\tau_0$  via equation (43).

$$h_s(t) = \sum_{n=0}^{N_f-1} h(t - nT_f) \quad (40)$$

$$x_{\tau_0} = h_s(t - \tau_0 + T_s) + h_s(t - \tau_0) \quad (41)$$

$$y_{\tau_0} = h_s(t - \tau_0 + T_s) - h_s(t - \tau_0) \quad (42)$$

$$\hat{\tau}_0 = \arg \max_{\tau} E \left\{ \int_0^{T_s} x_{\tau_0}(t + \tau) y_{\tau_0}(t + \tau) dt \right\} = \left| \int_0^{T_s + (\tau - \tau_0)} h_s^2(t) dt - \int_{T_s + (\tau - \tau_0)}^{T_s} h_s^2(t) dt \right| \quad (43)$$

[0071] FIG. 5 is a timing diagram illustrating an example waveform 40 with pairs of symbols alternating between positive and negative polarity which can be used as a training sequence for data-aided timing synchronization with dirty templates in communication system 2. Waveform 40 includes  $N_f=3$  received pulses with duration  $T_f$  per symbol which are represented by triangles to illustrate the multipath effects of channel 6 time spreading the energy of a transmitted pulse. The pulses of waveform 40 are also subject to time-hopping within each symbol interval of duration  $T_s$  42A-42E. Waveform segments 44A-44D are of symbol duration  $T_s$  and are expressed in terms of waveforms 32 and 34 with positive pulses illustrated having solid fill, while negative pulses are illustrated having a cross-hatch fill.

The timing offset created by the difference in time between the beginning of interval 36 and start of the next symbol interval is given by  $\tau_0 = n_\epsilon T_f + \epsilon$ .

[0072] Waveform 40 is designed according to equation (25) such that pairs of positive symbols and pairs of negative symbols are transmitted alternately. Consequently, equation (43) is used to estimate timing offset  $\tau_0$ .

[0073] FIGS. 6-8 are timing diagrams that illustrate an exemplary mode of operation of timing synchronization unit 14 employing data-aided timing synchronization with dirty templates as designed in FIG. 5. Transmitted waveform 50 includes  $N_f=3$  pulses per symbol represented by rectangles to illustrate the impulse-like nature of waveform 50. Alternating pairs of positive polarity and negative polarity pulses are transmitted as designed in equation (23). As in FIG. 5, positive symbols with are illustrated having solid fill, while negative pulses are illustrated having a cross-hatch fill.

[0074] Waveform 50 transmits symbols such that each segment 54A-54E has duration  $T_s$  and defines a discrete information-bearing symbol. Waveform 50 propagates through channel 6 and is received by receiver 8 as waveform 52. The triangular pulses of waveform 52 includes pulses illustrating the multipath effects of channel 6 which spreads the energy of the transmitted pulse in time. Although time-hopping is not illustrated in FIGS. 6-8, employing time-hopping does not create any limitations. Receiver 8 estimates timing offset  $\tau_0$  by operating on symbol long segments 56A-56D of waveform 52. Symbols 58A, 58B indicate cross correlation according to equation (44).

[0075] FIG. 6 illustrates the first cross correlation of two pairs of successive segments (56A, 56B) and (56C, 56D) of waveform 52. Equation (44) is used to calculate the averaged absolute value of cross correlation of the successive segments.  $M=3$  represents the 4 symbol duration segments of waveform 52 and  $n=0$  represents the number of increments. Consequently, equation (44) represents a value of  $E_h$ .

$$\bar{I}(n) := \frac{1}{M} \sum_{m=0}^{M-1} \left| \int_0^{T_s} x_{\tau_0}^{(m)}(t + nT_f) y_{\tau_0}^{(m)}(t + nT_f) dt \right| \quad (44)$$

[0076] FIG. 7 illustrates the cross correlation of successive segments (56A, 56B) and (56C, 56D) of waveform 52 after receiver 8 increments the starting time of the successive segments by one frame duration  $T_f$ . Equation (44) outputs a value of  $E_h$  for  $M=4$  and  $n=1$ .

[0077] FIG. 8 illustrates the cross correlation of successive segments (56A, 56B) and (56C, 56D) of waveform 52 after receiver 8 increments the starting time of the successive segments by one frame duration  $T_f$ . Equation (44) outputs a value of  $3 E_h$  for  $M=4$  and  $n=2$  because all the pulses in each symbol long segment are of the same polarity and of opposite polarity of the segment with which it is correlated. This result is consistent with Table 1. Equation (45) outputs the value  $n_e=2$  by peak-picking equation (44) across  $n$ .

$$\hat{n}_e = \arg \max_n E \left\{ |\bar{I}(n)| \right\} \quad (45)$$

[0078] FIG. 9 is a flow chart illustrating an example mode of operation of timing synchronization unit 14 of receiver 8 using data-aided timing synchronization with dirty templates. For each received burst,  $n$  is set equal to zero (step 60) so that receiver 8 initiates timing synchronization as soon as received waveforms are detected.  $M$  successive segments, each of duration  $T_s$ , are taken from the received waveform (step 62).  $M/2$  successive pairs are then cross correlated (step 64) and timing synchronization unit 14 increments  $n$  by one frame duration (step 67) based on whether successive segments have been incremented by a full symbol duration (step 66), i.e.,  $N_f$  iterations. In some embodiments  $n$  may be selected to acquire any desirable resolution. When the segments have been cross correlated over  $N_f$  successive frame durations, an estimate of  $n_e$  is formed by peak-picking the averaged cross correlations (step 68). The timing offset is estimated by correcting  $t_3$  with  $n_e T_f$  (step 69).

[0079] FIG. 10 illustrates the possible successive symbol long waveform segments 72-76 of a received waveform of communication system 2 employing non-data-aided timing synchronization with dirty templates. Waveforms 72-76 can be viewed as combinations of segments defined according to equations (41, 42). For example, the successive waveforms of

70 include only positive pulses resulting in a cross correlation given by  $\left| \int_0^{T_s} x_{\tau_0}^2(t) dt \right|$ .

Although  $x_{\tau_0}(t)$  is dependent on  $\tau_0$ , the auto-correlation of  $x_{\tau_0}(t)$  is independent of  $\tau_0$ .

Therefore, the auto-correlation of  $x_{\tau_0}(t)$  is a constant in time. Equation (46) gives the average of the absolute value of the cross correlations for successive segments 70-76 with equi-probable independent identically distributed symbols. As will be shown in greater detail below through simulations, equation (46) is used in the same manner as equation (44) to provide estimates of the timing offsets. The second term of equation (46) enables non-data-aided timing synchronization with dirty templates. In this case, received information symbols are correlated with received information symbols, resulting in an increase in energy capture because the successive waveform segments propagate through the same communication channel. Consequently, there is an increase in bandwidth efficiency but a decrease in performance. Non-data-aided timing synchronization with dirty templates typically requires correlation over a greater number of successive segments to achieve the same performance as data-aided timing synchronization with dirty templates.

$$E \left\{ \left| \text{successive correlations} \right| \right\} = \frac{1}{2} \left| \int_0^{T_s} x_{\tau_0}^2(t) dt \right| + \frac{1}{2} \left| \int_0^{T_s} x_{\tau_0}(t) y_{\tau_0}(t) dt \right| \quad (46)$$

[0080] FIG. 11 is a timing diagram that illustrates exemplary waveforms when communication system 2 employs data-aided timing synchronization with dirty templates in the presence of multiple users. The desired user's waveform 80 uses the training sequence given in equation (23). Successive segments of symbol duration  $T_s$  82 within waveform 80 are guaranteed to alternate between segments of  $x_{\tau_0}(t)$  and  $y_{\tau_0}(t)$ . However, the interferer's waveform 84 is not using the training sequence and is not guaranteed to alternate between segments of  $x_{\tau_0}(t)$  and  $y_{\tau_0}(t)$ . Consequently, successive segments 86 of the interferer's waveform result in cross correlation equations (47, 48).

$$\pm \int_0^{T_s} x_{\tau_0}^2(t) dt \quad (47)$$

$$\pm \int_0^{T_s} x_{\tau_0}(t) y_{\tau_0}(t) dt \quad (48)$$

**[0081]** By taking the absolute value of the averaged cross correlations, the interferer's waveform is eliminated, leaving only the desired user as shown in equation (49).

$$\bar{I}(n) := \frac{1}{M} \left| \sum_{m=0}^{M-1} \int_0^{T_s} x_{\tau_o}^{(m)}(t + nT_f) y_{\tau_o}^{(m)}(t + nT_f) dt \right| \quad (49)$$

**[0082]** FIGS. 12 and 13 are graphs illustrating exemplary simulation results from application of equations (27, 28) with and without time-hopping codes respectively. Over 100 channel realizations are plotted for each equation. Both  $\epsilon_0$  and time-hopping codes are randomly generated according to a uniform distribution on intervals  $[0, T_s)$  and  $[0, N_c-1]$  with  $N_c=90$  respectively. Equations (27, 28) do not vary significantly with and without time-hopping codes for various values of  $\epsilon_0$  and channel realizations.

**[0083]** FIGS. 14-17 are graphs illustrating results of simulations and comparisons that validate analyses and designs of data-aided and non-data-aided timing synchronization with dirty templates. The simulations are carried out with the pulse  $p(t)$  selected as the second derivative of the Gaussian function with unit energy and duration  $T_p \approx 1$  nanosecond (ns). Each symbol contains  $N_f=32$  frames, each with duration  $T_f=100$  ns. Random channels are generated where rays arrive in several clusters within an observation window. The cluster arrival times are modeled as Poisson variables with cluster arrival rate  $\Lambda$ . Rays within each cluster also arrive according to a Poisson process with ray arrival rate  $\lambda$ . The amplitude of each arriving ray is a Rayleigh distributed random variable having exponentially decaying mean square value with parameters  $\Gamma$  and  $\gamma$ . Parameters of this channel are chosen with  $(1/\Lambda, 1/\lambda, \Gamma, \gamma)=(2, 0.5, 30, 5)$  ns. The diminishing tail of the power profile is truncated to make the maximum delay spread of the multipath channel  $T_g=99$  ns. The signal-to-noise ratio (SNR) is defined as the energy per symbol (bit for binary PAM) over the noise variance and  $\epsilon$  is randomly generated from a uniform distribution over  $[0, T_s)$ . Fast TH spreading codes of period  $N_f$  are generated from a uniform distribution over  $[0, N_c-1]$  with  $N_c=90$  and  $T_c=1.0$  ns, independently from frame to frame. Inter-symbol interference is avoided by setting  $c_0=c_{N_f-1}=0$ .

**[0084]** FIGS. 14 and 15 are graphs illustrating results of a comparison of mean square error (MSE) performance of data-aided and non-data-aided timing synchronization with dirty

templates for various values of  $M$ , respectively. The MSE in FIGS. 14 and 15 are normalized with respect to  $T_s^2$  and plotted verse the signal-to-noise ratio (SNR) per pulse. Also plotted is the normalized variance of the random timing offset without timing synchronization. In general, as  $M$  increases the normalized MSE decrease monotonically. Similarly, the MSE curves flatten as SNR increases of the frame resolution limit.

[0085] Data-aided timing synchronization techniques have higher performance than the non-data-aided techniques. For example, data-aided timing synchronization of FIG. 14 is operational with  $M=2$ , i.e. two pairs of  $x_{\tau_0}(t)$  and  $y_{\tau_0}(t)$ , and performs significantly better than the non-data-aided  $M=4$  case of FIG. 15. In general, non-data-aided techniques with values of  $M \geq 16$  result in MSE performance similar to the performance of data-aided techniques with  $M/4$  training symbols.

[0086] FIGS. 16 and 17 are graphs illustrating results of a comparison of bit-error-rate (BER) performance of data-aided and non-data-aided timing synchronization with dirty templates for various values of  $M$ , respectively. Also plotted are the perfect timing case and the case without timing synchronization. The channel estimate is free of errors in order to isolate timing from channel estimation. The BER performance increases monotonically as  $M$  increases in both FIG. 16 and 17. FIG. 16 shows that data-aided timing synchronization techniques are operational with  $M=2$ , i.e. two pairs of  $x_{\tau_0}(t)$  and  $y_{\tau_0}(t)$ , while the non-data-aided techniques require  $M=32$  to be operational.

[0087] The described techniques may provide advantages in UWB communications systems, and non-UWB wireless communication systems where inter-symbol interference is absent or substantially reduced. In particular, the described techniques operate in realistic UWB environments, e.g., environments having the presence of noise, multipath, time-hopping, and multiple users. Timing synchronization with dirty templates peak-pick the output of a sliding correlator with a template that propagates through the wireless communication channel. Because the template propagates through the same communication channel, the resulting energy capture is higher than energy capture using a clean template, i.e., a stored segment of the transmit waveform. Selecting a data-aided, dirty template increases the performance of the described techniques. For example, a dirty template with four symbols increases the performance by reducing the number of operations performed at the receiver. However, selecting a blind, i.e. non-data-aided, template provides a more efficient use of bandwidth.



Additionally, timing synchronization can be achieved at any desirable resolution by selecting the increment at which each correlation is performed within a symbol duration. In some embodiments, voltage controlled clock (VCC) circuits can be used to implement the receiver. [0088] The described techniques can be embodied in a variety of devices that communicate using ultra wideband communication, including base stations, mobile phones, laptop computers, handheld computing devices, personal digital assistants (PDA's), a device within a wireless personal area network, a device within a sensor network, and the like. The devices may include a digital signal processor (DSP), field programmable gate array (FPGA), application specific integrated circuit (ASIC) or similar hardware, firmware and/or software for implementing the techniques. If implemented in software, a computer-readable medium may store computer readable instructions, i.e., program code, that can be executed by a processor or DSP to carry out one of more of the techniques described above. For example, the computer-readable medium may comprise random access memory (RAM), read-only memory (ROM), non-volatile random access memory (NVRAM), electrically erasable programmable read-only memory (EEPROM), flash memory, or the like. The computer-readable medium may comprise computer readable instructions that when executed in a wireless communication device, cause the wireless communication device to carry out one or more of the techniques described herein. These and other embodiments are within the scope of the following claims.